24th Annual
Software Engineering Workshop
December 1999

Capability Maturity Model
Level 4
Quantitative Analysis

Al Florence

The views expressed are those of the author and do not reflect the official policy or position of the MITRE Corporation

Agenda

- Capability Maturity Model (CMM) Level 4/5 overview
- Level 4
  - Quantitative Process Management (QPM)
  - Software Quality Management (SQM)
  - Statistical process control (SPC)
  - Quantitative analysis
- Level 5
  - Defect Prevention (DP)

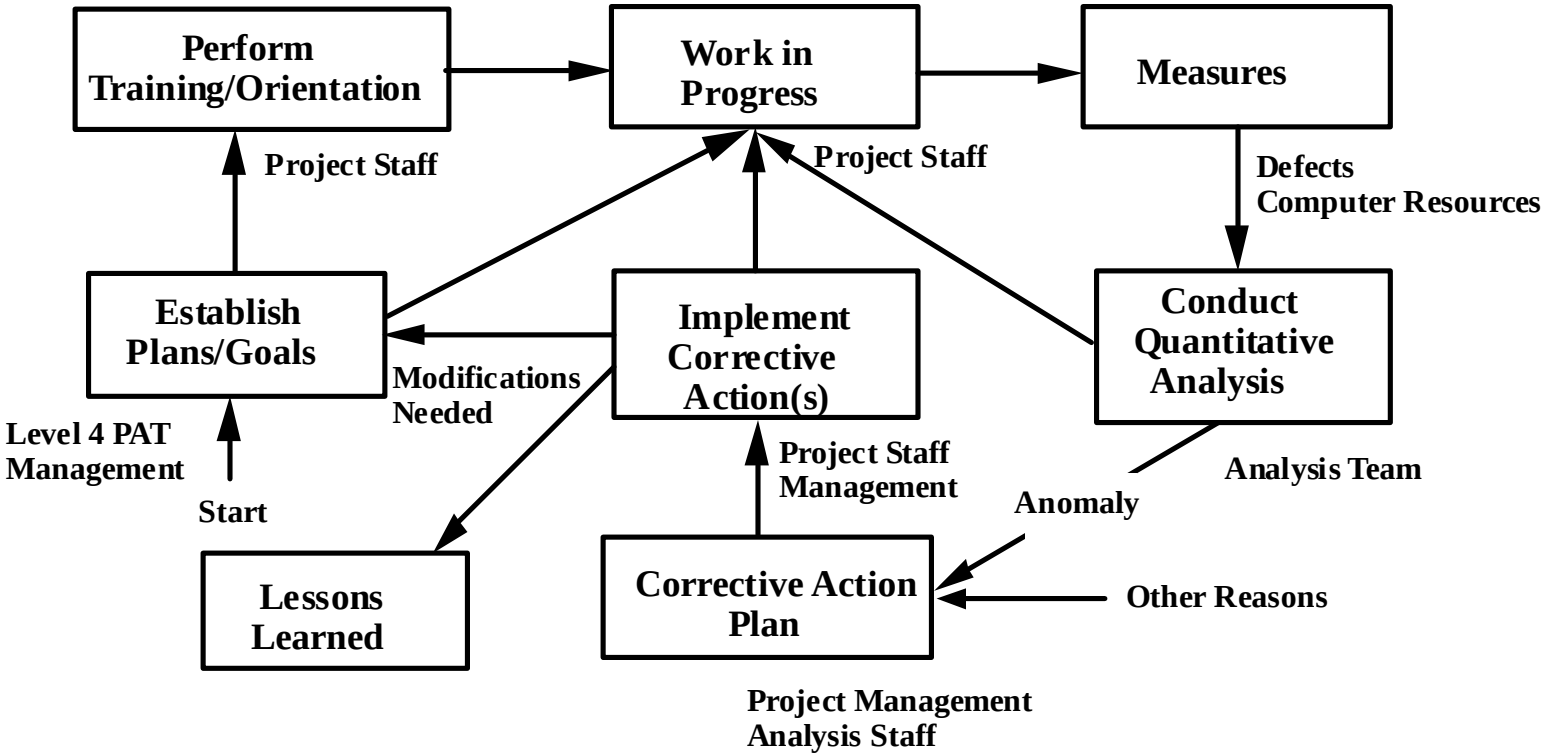
Level 4 - Managed

- **Quantitative Process Management - Process Focus**
  - **To control the process performance of the software project quantitatively**
- **Software Quality Management - Product Focus**
  - **To develop a quantitative understanding of the quality of the project's software products and achieve specific quality goals**

Level 5 Optimizing

- **Defect Prevention**
  - **To identify the cause of defects and prevent them from recurring**
- **Technology Change Management**
  - **To identify new technologies (i.e., tools, methods, and processes) and transition them into the organization in an orderly manner**
- **Process Change Management**
  - **To continually improve the software process used in the organization with the intent of improving software quality, increasing productivity, and decreasing the cycle time for product development**

Level 4 QPM/SQM Process



PAT - Process Action Team

Level 4 Plans/Goals

- **Level 4 goals, and plans to meet those goals, are based on the project's proven capability to perform**
- **Goals and plans must also reflect contract requirements**
- **As the project's capabilities and contract requirements change, the goals and plans may need to be adjusted**

Plans/Goals Example - Actual Project

- **Project's key driving requirements**
  - **Timing - subject search response in less than 2.8 seconds 98% of time**
  - **Availability - 99.86% 7 days, 24 hours (7/24)**
- **These are driving requirements that constrain hardware & software architecture & design**
- **To satisfy these requirements, the system needs to be highly reliable and hardware robust**
- **The Planned Quality Goals are:**
  - **Deliver a near defect free system**
  - **Meet all Critical Computer Performance Goals**

Plans/Goals Example

- **Plans are to detection and removal defects during:**
  - **Requirements peer reviews**
  - **Design peer reviews**
  - **Code peer reviews**
  - **Unit tests**
  - **Thread tests**
  - **Integration and test**
  - **Formal test**

Plans/Goals Example

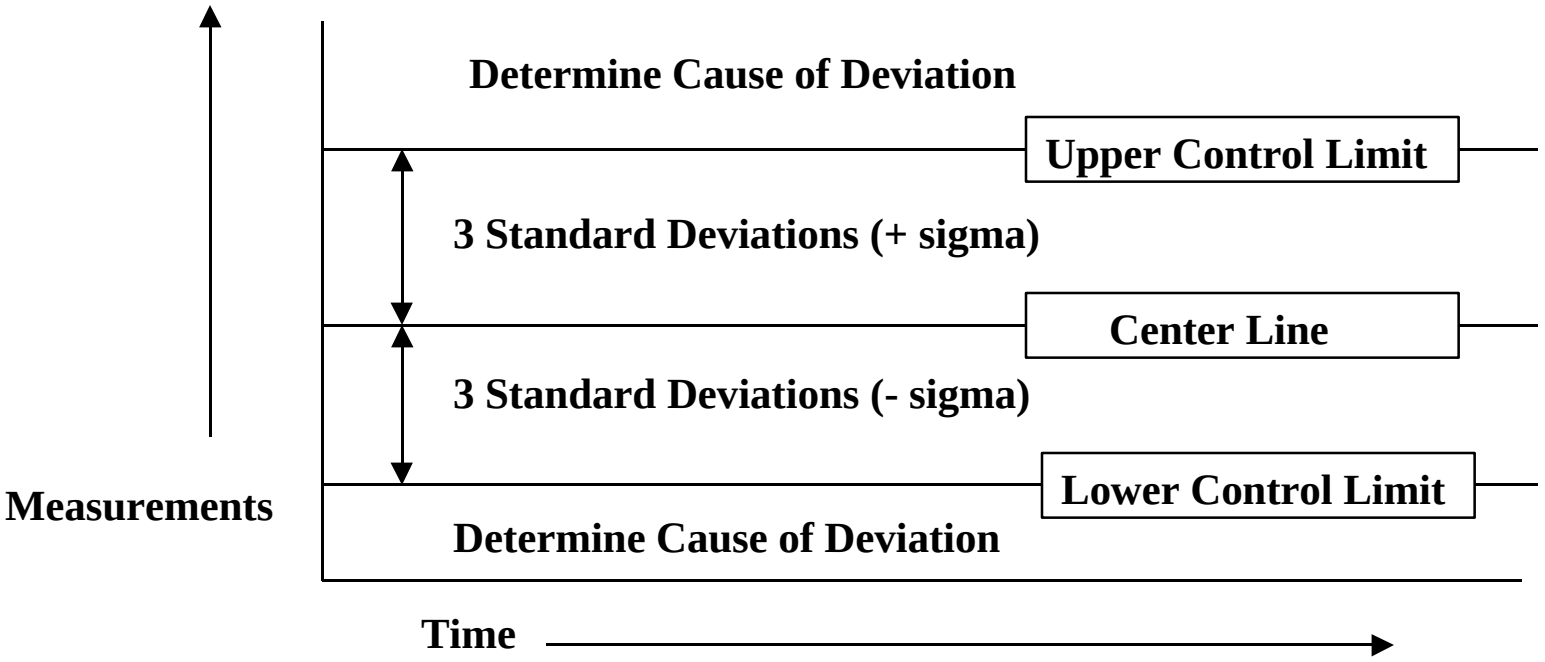
- **Plans are to monitor Critical Computer Resources**
  - **General Purpose Million Instructions Per Second (MIPS)**
  - **Disc Storage Read Inputs/Outputs Per Second (IOPS) Per Volume**
  - **Write IOPS Per Volume**
  - **Operational Availability**
  - **Peak Response Time**
  - **Server Loading**

Statistical Tools

The following tools were used to conduct the quantitative analysis

- **Statistical Process Control (SPC) - SPC using control charts & Bar Charts**
- **Performance Model - To monitor critical computer resource**

Statistical Process Control Charts



According to the Normal Distribution, 99% of all normal random values lie within +/-3 standard deviations from the norm, that is, 3 sigma

Statistical Process Control Charts

Why Control charts:

- **Separate signal from noise, so when anomalies occur they can be recognized**
- **Identify undesirable trends, they point out:**
  - **Fixable problems**
  - **Potential process improvements**
- **Show the capability of the process, so achievable goals can be set**
- **Provide evidence of process stability, which justifies predicting process performance**

Variables Data and Attributes Data

- **Variables Data**
  - **Usually measurements of continuous phenomena**
    - **Length, weight, height, volume, voltage, torque**
  - **In software settings**
    - **Elapsed time, effort expanded, memory/cpu utilization**
- **Attributes Data**
  - **Usually measurements of discrete phenomena (counts)**
    - **Number of defects, number of source statements, number of people**
  - **Most measurements in software used for SPC are attributes data**

Variables Data and Attributes Data

- **Control Limits**
  - **Control limits for variables and attributes data are computed in quite different ways**
- **Control Charts for Variables Data**
  - **X bar**
  - **Range charts**
  - **XmR Charts**
- **Control Charts for Attributes Data**
  - **u charts**
  - **Z charts**
  - **XmR Charts**

Other Quantitative Methods

- Check Sheets
- Run Chart
- Histogram
- Scatter Diagram
- Pareto Chart
- Flow Chart
- Fishbone Diagram

SPC Example - Code Peer Reviews

Raw data collected for code peer reviews

Sample	*Units*	*SLOC*	*Defects*	*Defects/KSLOC*
1. Feb 1997	17	1705	62	36.36
2. Mar 1997	18	1798	66	36.71
3. Mar 1997	15	1476	96	65.04
4. Mar 1997	19	1925	57	29.61
5. Mar 1997	17	1687	78	46.26
6. Apr, May	18	1843	66	35.81
Totals	104	10434	425	

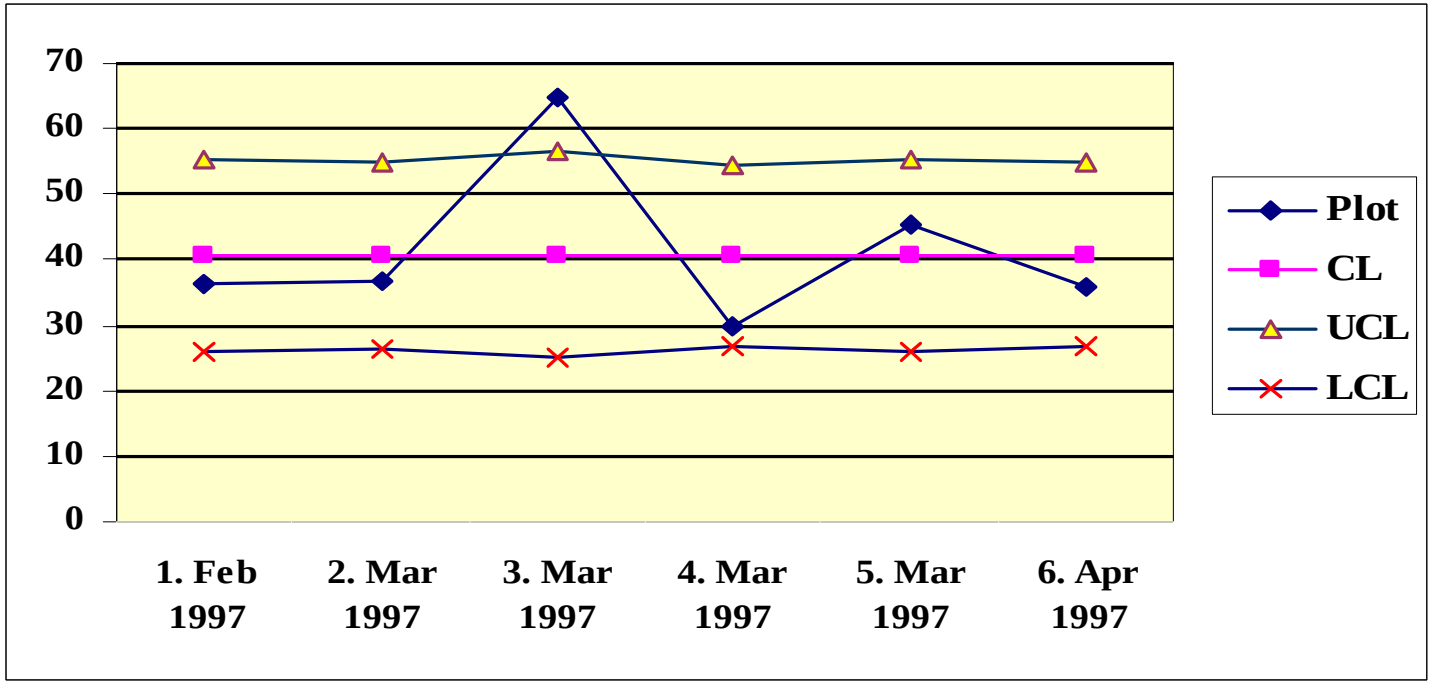
Calculating the limits

- **Defects/KSLOC = Number of Defects*1000/SLOC reviewed per sample (calculated for each sample). These are plotted as Plot.**
- **CL = Total Number of Defects/Total number of SLOC reviewed * 1000**
- **a(1) = SLOC reviewed/1000 (calculated for each sample)**
- **UCL = CL+3(SQRT(CL/a(1)) (calculated for each sample)**
- **LCL = CL-3(SQRT(CL/a(1)) (calculated for each sample)**

Calculations for each sample

Sample	Plot	CL	UCL	LCL	a(1)
1. Feb 1997	36.4	40.73	55.4	26.09	1.7
2. Mar 1997	36.7	40.73	55.01	26.45	1.8
3. Mar 1997	65	40.73	56.49	24.97	1.5
4. Mar 1997	29.6	40.73	54.53	26.93	1.9
5. Mar 1997	45.2	40.73	55.47	25.99	1.7
6. Apr 1997	35.8	40.73	54.84	26.63	1.8

Code Peer Reviews Control Chart

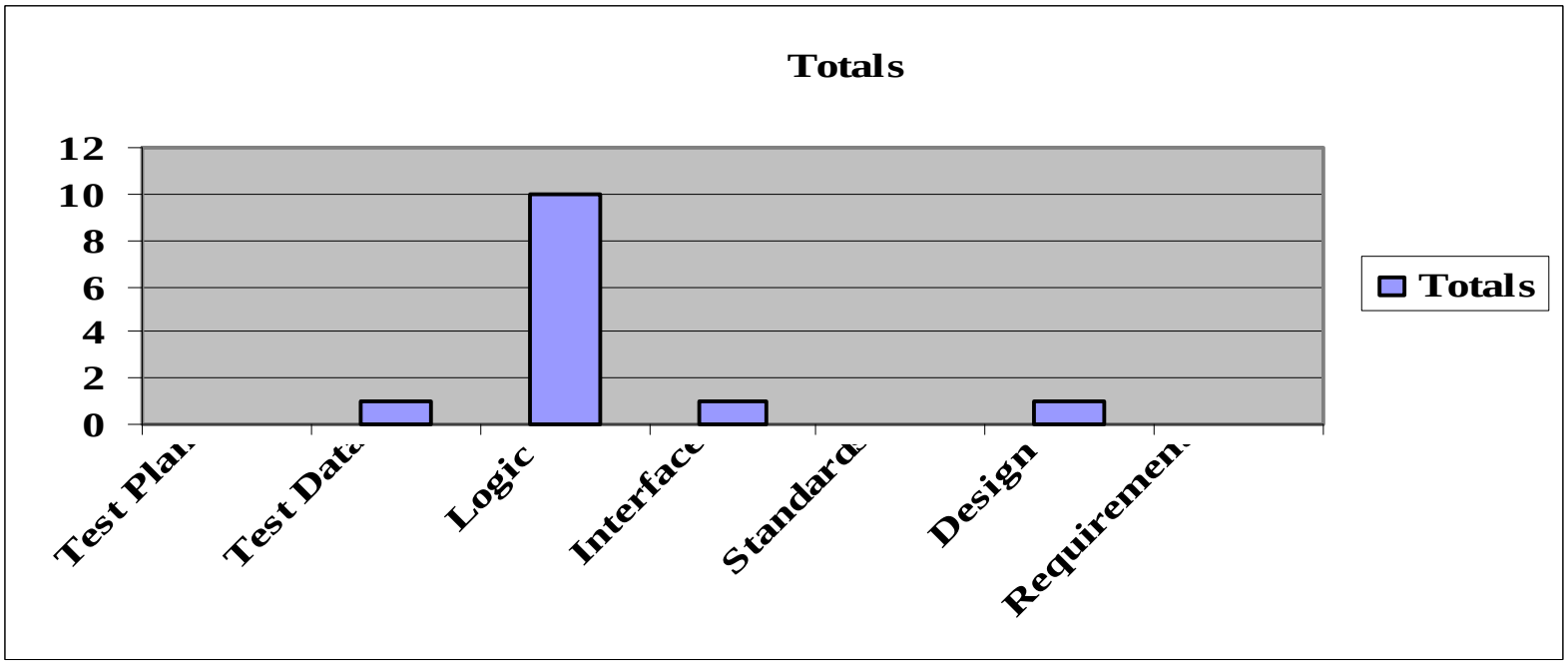


- The process is out of statistical process control in the third sample
- Analysis revealed that this was caused when the project introduced coding standards and many coding violations were introduced

MITRE

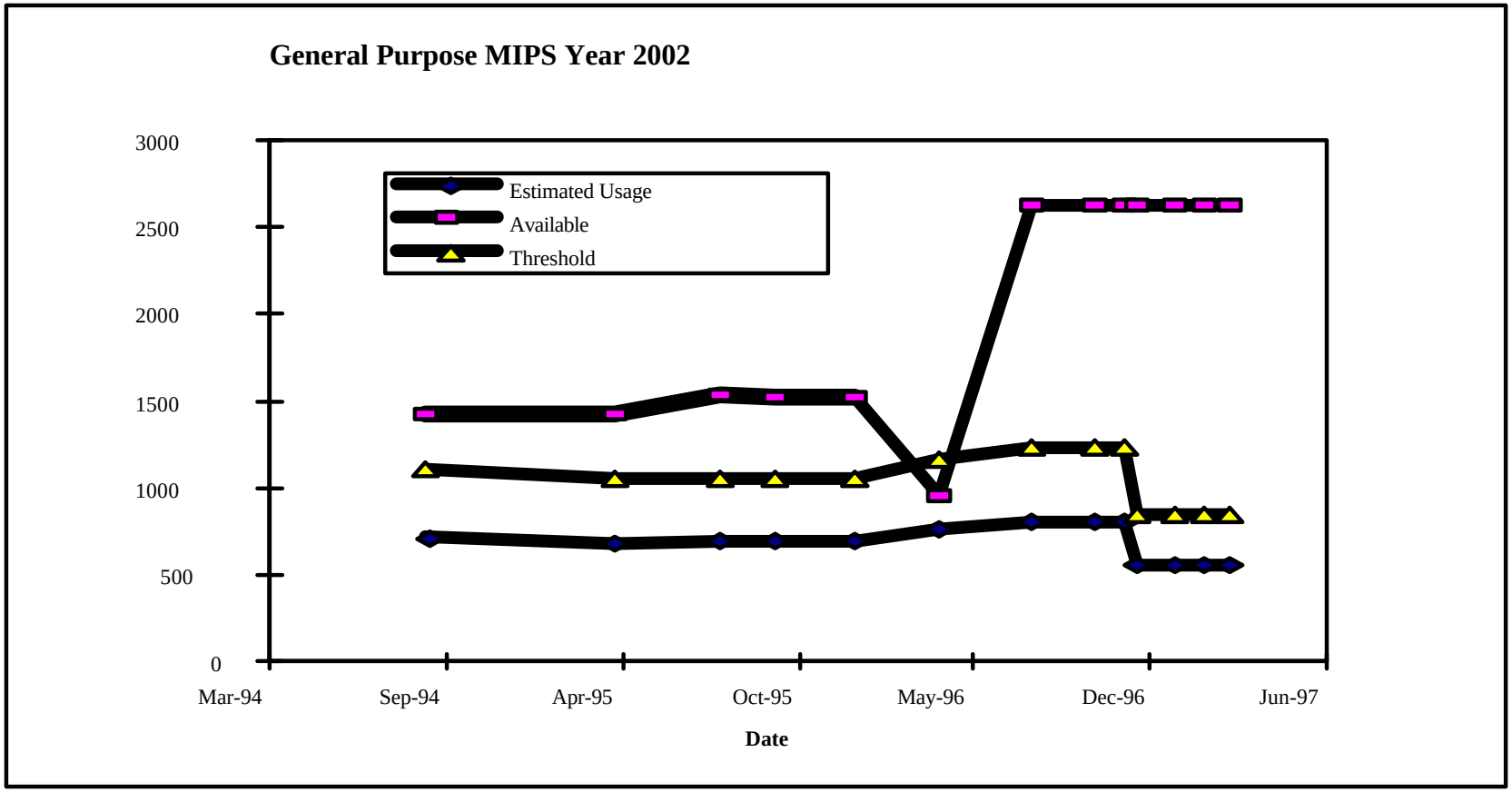
Bar Charts for Thread Tests

Samples	Test Plan	Test Data	Logic	Interface	Standards	Design	Requirements	-	Defect Categories
1			6	1		1			
2		1							
3			4						
Totals	0	1	10	1	0	1	0		



You would expect more logic defects than others

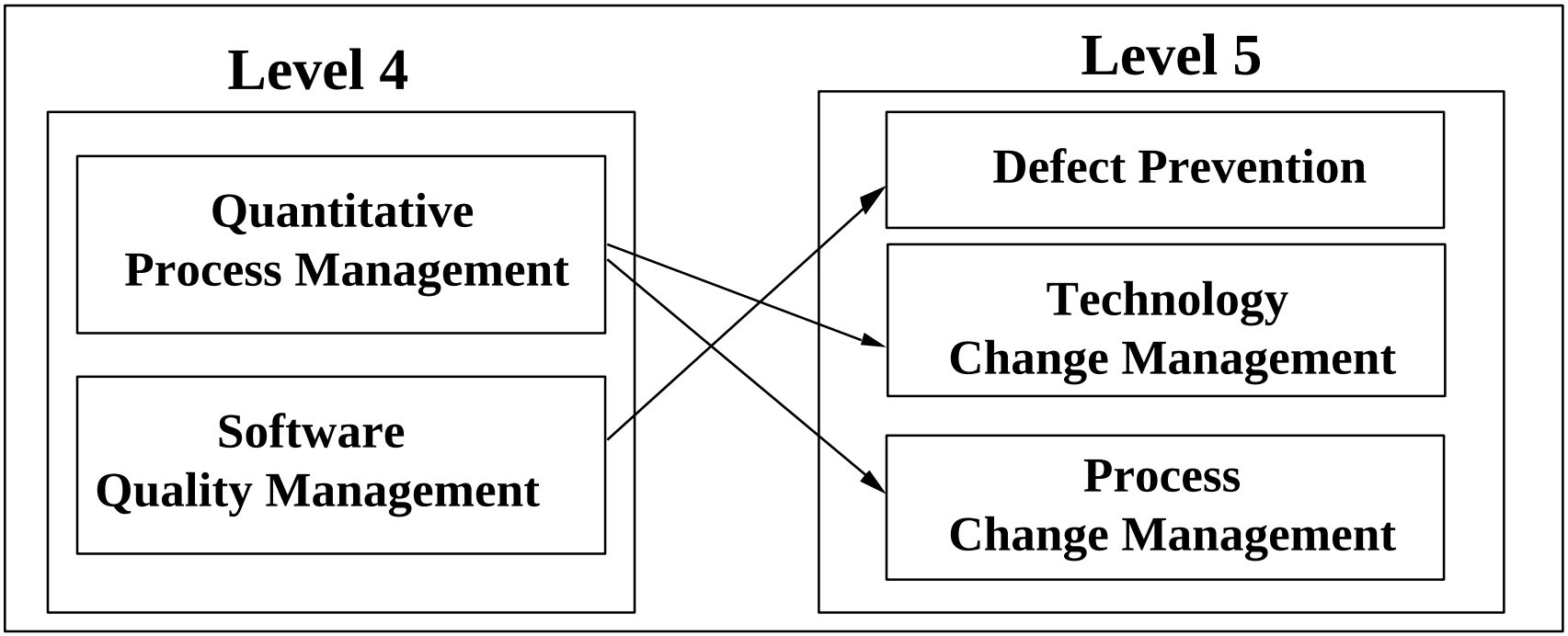
Example
Critical Computer Resources



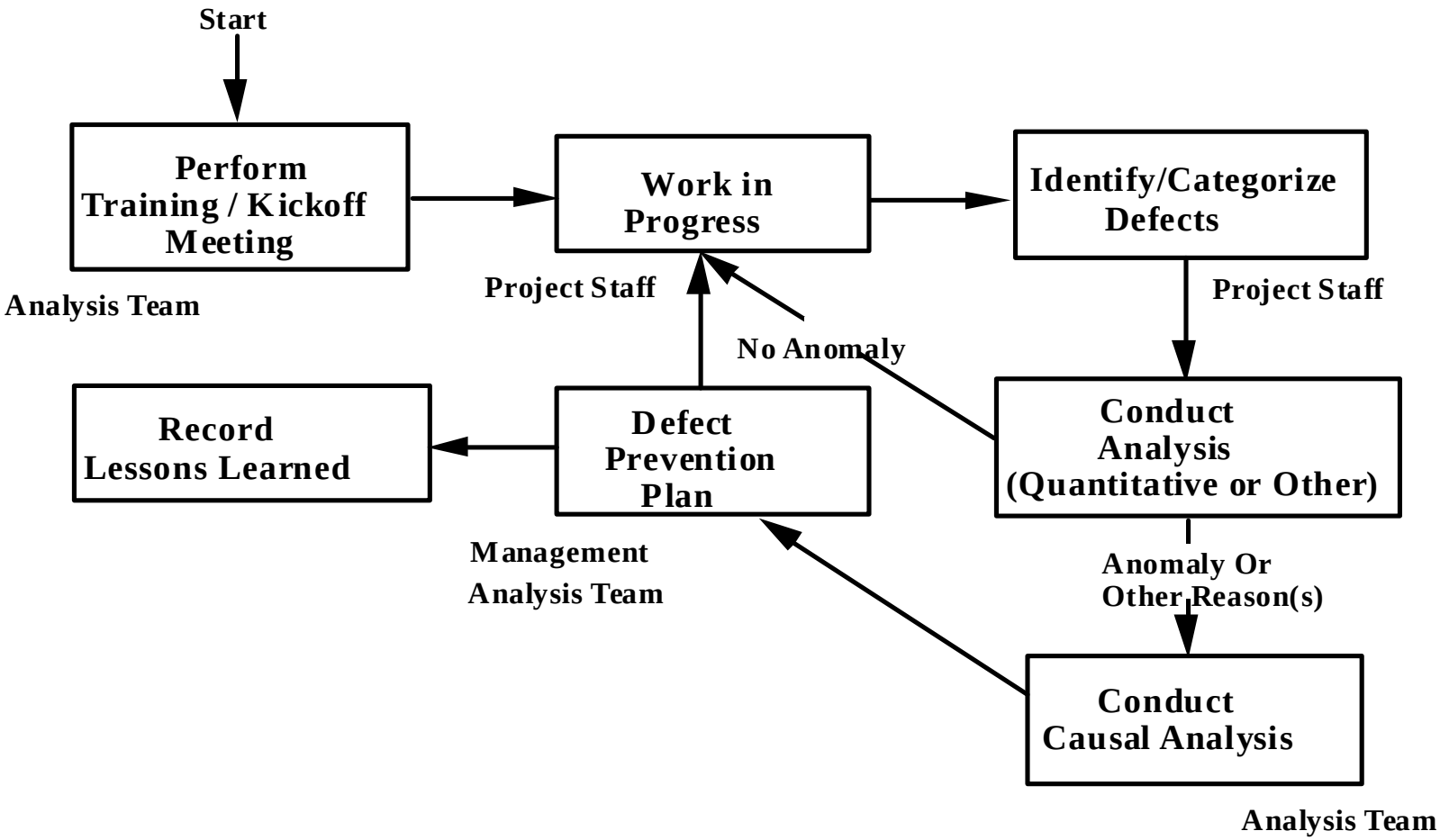
- **The customer introduced many new requirements around Nov/Dec 1995**
- **The model revealed that the MIPS threshold was threatened with increased computations**
- **More MIPS were added to the architecture in May 1996**

Level 4 to Level 5 Relationships

Data analysis at Level 4 enables focusing the on Defect Prevention, Technology Change Management, and Process Change Management at Level 5



Defect Prevention Process



Defect Prevention Plans and Activities

- **Defects can be prevented on a variety of entities:**
  - **Project Plans**
  - **Project Resources**
  - **Quality Goals**
  - **Design**
  - **Interfaces**
  - **Test Procedures**
  - **Processes**
  - **Technologies**
  - **Management**
  - **Project Schedules**
  - **Standards**
  - **Requirements**
  - **Code**
  - **Test Plans**
  - **Documentation**
  - **Procedures**
  - **Training**
  - **Engineering**

Defect Prevention Example

Raw Data - Code Peer Review

Sample	Units	SLOC	Defects	Defects/KSLOC
1. Mar 1998	6	515	15	29.12621359
2. Apr 1998	10	614	16	26.05863192
3. Apr 1998	7	573	7	12.21640489
4. Apr 1998	7	305	7	22.95081967
5. Apr 1998	4	350	21	60
6. Apr 1998	3	205	2	9.756097561
7. Apr 1998	8	701	11	15.69186876
8. May 1998	3	319	3	9.404388715
Totals	76	3582	72	

Defect Prevention Example (Cont.)

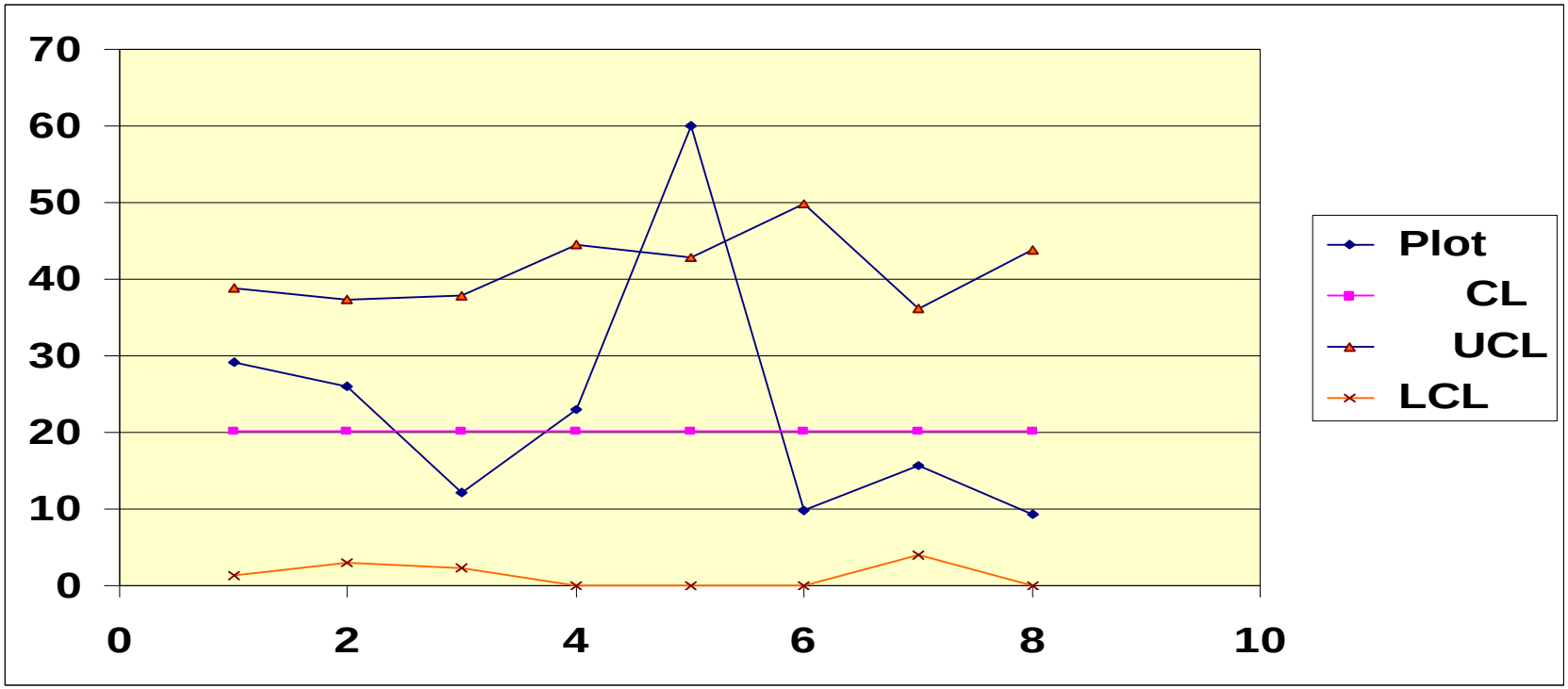
Calculations

Sample	Plot	CL	UCL	LCL	a(1)
1. Mar 1998	29.1	20.1	38.843	1.358279654	0.515
2. Apr 1998	26.1	20.1	37.265	2.935632196	0.614
3. Apr 1998	12.2	20.1	37.869	2.332140203	0.573
4. Apr 1998	23	20.1	44.455	0	0.305
5. Apr 1998	60	20.1	42.835	0	0.35
6. Apr 1998	9.76	20.1	49.807	0	0.205
7. Apr 1998	15.7	20.1	36.165	4.036058356	0.701
8. May 1998	9.4	20.1	43.914	0	0.319

When LCL is negative it is set to zero

Defect Prevention Example (Cont.)

Plot

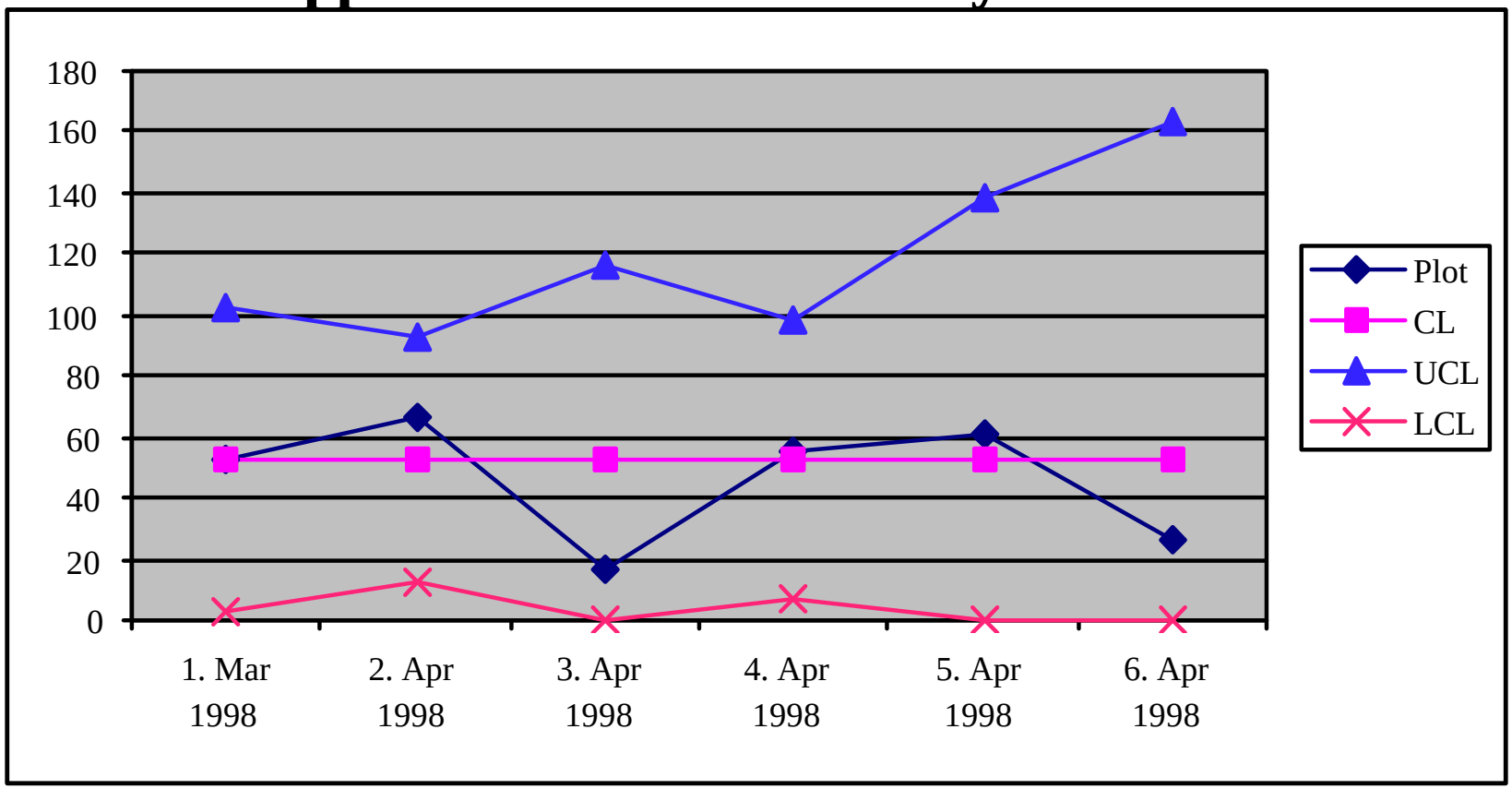


Causal Analysis

- Revealed that data were for database code and applications code
- Control charts require similar data for similar processes
- Apples to apples analogy

Defect Prevention Example (Cont.)

Plot - Applications Code Data Only



Process is now under statistical process control

Defect Prevention Example (Cont.)

Root Cause

- **Data gathered from dissimilar activities cannot be used on the same statistical process on control charts**
- **Data from design cannot be combined with data from coding**
- **The process for database design and code is different from that used for applications design and code as are the teams and methodologies**

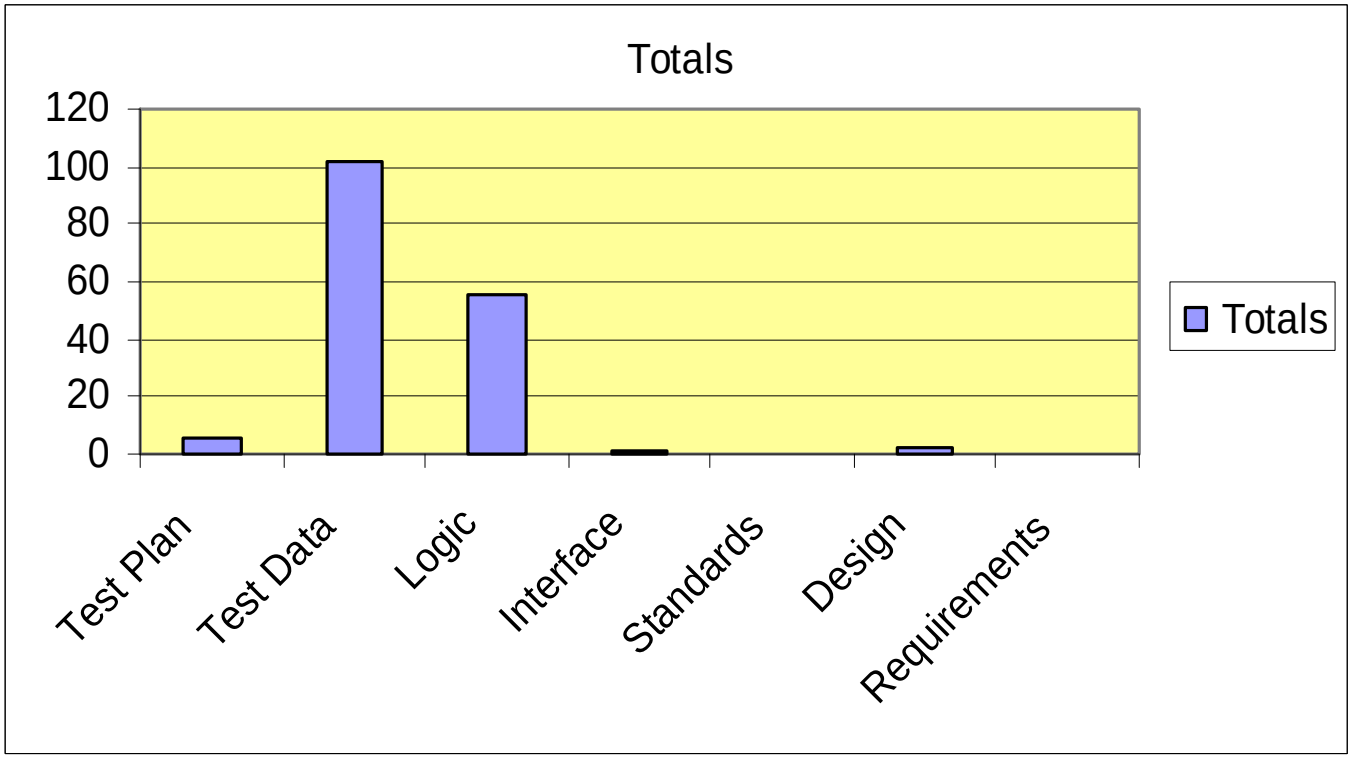
Defect Prevention

The defect prevention is against the process of collecting data for SPC control charts

Thread Tests

Samples	Test Plan	Test Data	Logic	Interface	Standards	Design	Requirements
1	2	6					
2		10					
3	1	9	3				
4	2	1	13				
5		1	7				
6		10	14				
7		4	2				
8		28					
9						2	
10			6				
11	1	1					
12		10					
13		9	1				
14		6	1	1			
15		5	7				
16		2	1				
Totals	6	102	55	1	0	2	0

Bar Chart for Thread Tests



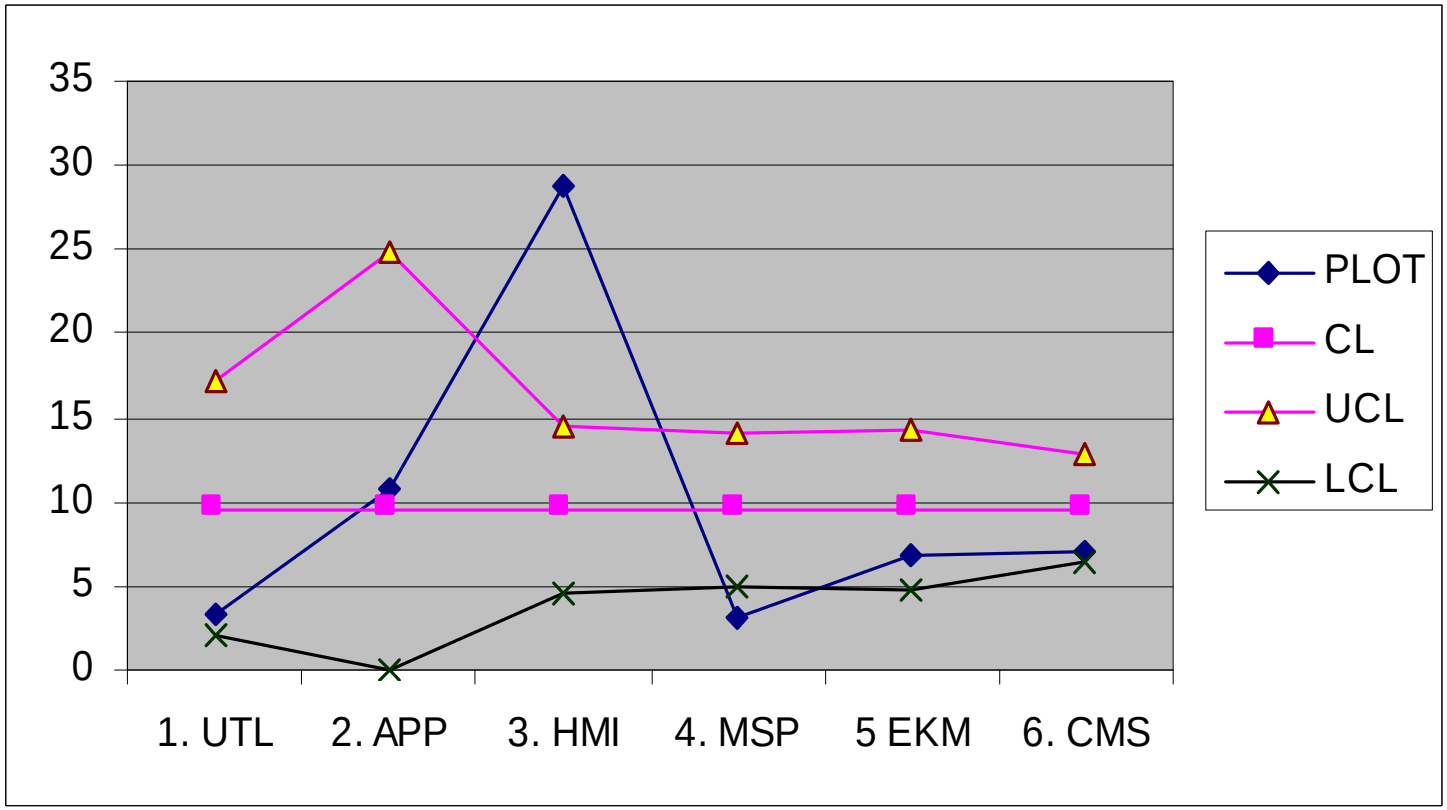
- **Test data would not be expected to have the majority of defects**
- **The root cause is that test procedures had not been peer reviewed**
- **The defect prevention is to peer review test procedures**

Requirements Defects

Sample	SRSs	No. Rqmts	Defects	Defects/100 Rqmts
1. UTL	1	152	5	3.28
2. APP	1	37	4	10.81
3. HMI	1	350	101	28.85
4. MSP	1	421	13	3.08
5. EKM	1	370	25	6.757
6. CMS	1	844	60	7.10
Totals	6	2174	208	

Sample	PLOT	CL	UCL	LCL	a(1)
1. UTL	3.28	9.56	17.09	2.04	1.52
2. APP	10.81	9.56	24.82	0	0.37
3. HMI	28.85	9.56	14.52	4.60	3.5
4. MSP	3.08	9.563	14.09	5.04	4.21
5. EKM	6.75	9.563	14.39	4.74	3.7
6. CMS	7.10	9.56	12.76	6.37	8.44

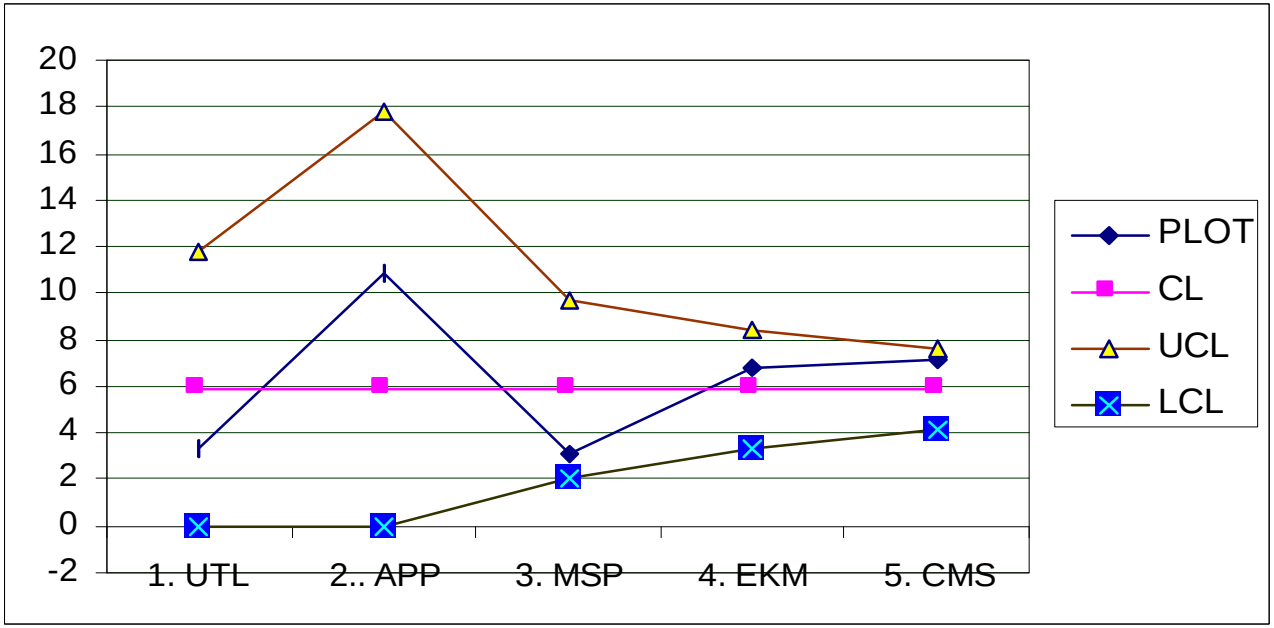
Requirements Defects (Cont.)



- **HMI is Human Machine Interface, Others are Applications**
- **Again, dissimilar activities cannot be used on the same statistical process on control charts**

Requirements Defects (Cont.)

Sample	SRSs	No. Rqmts	Defects	Defects/100 Rqmts
1. UTL	1	152	5	3.29
2. APP	1	37	4	10.81
3. MSP	1	421	13	3.09
4. EKM	1	370	25	6.76
5. CMS	1	844	60	7.11
Totals	5	1824	107	



Without HMI

References

- **Software Engineering Institute *Capability Maturity Model* (SEI CMM) V1.1,** 1993
- ***Software Measures and the Capability Maturity Model,* John H. Baumert, Mark S. McWhinney, SEI, 1992**
- ***Practical Software Measurement: Measuring for Process Management and Improvement,* William A. Florac, Robert E. Park, Anita D. Carleton, SEI, April, 1997**
- ***An SPC Primer*, Quality America, Inc., 1984, Thomas Pyzdek**
- ***Getting to Level 4 in the CMM*, Ron Radice, SEPG 97, San Jose, CA, 1997**
- ***Statistical Process Control for Software*, Anita Carleton, Mark C. Paulk, SEI, The 97 Software Engineering Symposium, 1997**
- ***Understanding Statistical Process Control*, David S. Chambers & Donald J. Wheeler, SPC Press, 1995**
- ***Juran's Quality Control Handbook*, 4th Edition, McGraw-Hill Book Company, 1988**
- ***Understanding Variation, The Key to Managing Chaos*, Donald J. Wheeler, SPC Press, 1993**
- ***Advanced Topics in Statistical Process Control*, Donald J. Wheeler, SPC Press, 1995**
- ***On Probability As a Basis For Action*, The American Statistician, Vol. 29, No.4 (146-152), Edwards W. Deming, November 1975**
- ***Assumptions for Statistical Inference*, The American Statistician, Vol. 47, No.1 (1-11), Gerald J. Hahn & William Q. Meeker, February 1993**
- ***Managing the Software Process*, Watts S. Humphrey, SEI Series in Software Engineering, Addison-Wesley Publishing Company, September 1997**

Contact Information

Alfred (Al) W. Florence

Florence@mitre.org

(703) 883-7476